# Nature Inspired Computing Techniques for Optimal Reactive Power Dispatch

Rebecca Ng Shin Mei,
Faculty of Electrical & Electronics Engineering,
University Malaysia Pahang,
26600 Pekan, Pahang, Malaysia,
rebeccamei527@gmail.com[*1]

Mohd Herwan Sulaiman,
Faculty of Electrical & Electronics Engineering,
University Malaysia Pahang,
26600 Pekan, Pahang, Malaysia,
herwan@ump.edu.my[2]

Zuriani Mustaffa,
Faculty of Computer System & Software Engineering,
University Malaysia Pahang,
26300 Gambang, Pahang, Malaysia,
zuriani@ump.edu.my[2]

Hamdan Daniyal,
Faculty of Electrical & Electronics Engineering,
University Malaysia Pahang,
26600 Pekan, Pahang, Malaysia,
hamdan@ump.edu.my[2]

*Abstract*—This paper presents the application of recent nature inspired computing techniques namely gray wolf optimizer (GWO) and ant lion optimizer (ALO) in solving optimal reactive power dispatch (ORPD) problem. GWO and ALO are utilized to minimize the transmission losses by finding the best combination of control variables such as generator voltages, transformer tap ratios as well as reactive compensation devices. In this paper, IEEE 30-bus system is utilized to show these techniques in solving ORPD. The comparison between the effectiveness of GWO and ALO are made and reported in this paper. The results show that GWO is able to gain a better result in solving ORPD than ALO.

*Keywords—Ant lion optimizer; Gray wolf optimizer; Loss minimization; Nature Inspired Computing Techniques; Optimal reactive power dispatch*

## 1. INTRODUCTION

Electrical power system is a network which comprising generation, transmission, distribution and supply the electrical power to the load. In latter developments, it is prospected to minimize the consumption of resources of power system, meanwhile maximizing the reliability as well as security. Undeniably, optimal reactive power dispatch (ORPD) acts as an important role in securing economic operation and electricity of power system. ORPD is a well-known nonlinear optimization problem in power system which including both the discrete and continuous control variables and at the same time satisfying the equality and inequality constraints. ORPD can be grouped as the sub problem of the optimal power flow (OPF) calculations which determines the control variables such as generator voltages, transformer tap ratios as well as shunt compensators meanwhile minimizes the transmission losses and other objective functions.

Undeniably, the ORPD problem had been reported in other literature and solved by using a numerous types of methods such as classical method including linear and non-linear programming, quadratic programming, interior point method, Newton method as well as gradient method [1], [2-6]. However, as compare to classical conventional method, recent development of nature inspired meta-heuristic techniques can achieve a better solutions in overcoming the ORPD problem. Moreover, various types of search techniques such as tabu search (TS), genetic algorithm (GA), evolutionary programming (EP) and evolutionary search (ES) have been implemented in solving ORPD problem but yield a poor result in solving optimization problem with discrete nature [7]. As a result, nature inspired meta-heuristic techniques have been exploited in overcoming the problem including artificial bee colony (ABC) [7], harmony search algorithm (HSA) [8], improved HSA [9], particle swarm optimization (PSO) [10],[11], gravitational search algorithm (GSA) [12],[13], gray wolf optimizer (GWO) [14], ant lion optimizer (ALO) [17] and many more.

In this paper, the application of nature inspired computing techniques which are gray wolf optimizer (GWO) and ant lion optimizer (ALO) are implemented in order to solve ORPD. This paper is organized as follows: Section 2 discusses the mathematical formulation of ORPD problem as well as a concise description of GWO and ALO methods are explained in Section 3. Section 4 introduces the implementation of GWO and ALO in solving ORPD followed by simulation results and discussion. Last but not least, Section 5 states the conclusion.

## 2. ORPD PROBLEM FORMULATION

*A. Objective Function*

The objective function of ORPD problem is to identify the total minimum transmission losses meanwhile fulfilling the equality and inequality constraints. The ORPD problem can be mathematically be formulated as follows:

Minimize $f(x,u)$

Subjected to

$$g(x,u) = 0$$
$$h(x,u) \leq 0 \tag{1}$$

where function $f(x,u)$ is the objective function, $g(x,u) = 0$ is the equality constraint as well as $h(x,u) \leq 0$ is the inequality constraint. Additionally, $x$ is the dependent variables vector and $u$ is the control variables vector. As mentioned above, the objective function, $f(x,u)$ of the ORPD problem is to find out the total transmission losses of the system at the same time satisfying both the equality constraint, $g(x,u) = 0$ and inequality constraint, $h(x,u) \leq 0$. The total transmission loss, $F$ is illustrated as follows [8]:

$$F = P_{Loss}(x,u) = \sum_{L=1}^{N_l} P_{Loss} \tag{2}$$

where $N_l$ is the number of transmission lines.

*B. Equality Constraint*

The equality constraint which is the power flow equalities stated in [8] declare that the total power generation must be equal to the total load demands plus total real power losses as expressed as below:

$$P_{Gi} - P_{Di} = V_i \sum_{j \in N_i} V_j (G_{ij} cos\theta_{ij} + B_{ij} sin\theta_{ij}) \tag{3}$$

$$Q_{Gi} - Q_{Di} = V_i \sum_{j \in N_i} V_j (B_{ij} cos\theta_{ij} - G_{ij} sin\theta_{ij}) \tag{4}$$

where $P_{Gi}$ and $Q_{Gi}$ are real and reactive power generation respectively, $P_{Di}$ and $Q_{Di}$ are real and reactive load demand respectively, $V_i$ and $V_j$ are voltage at load bus-i and bus-j respectively, $G_{ij}$ and $B_{ij}$ are the conductance and susceptance between bus-i and bus-j respectively.

*C. Inequality Constraint*

1. *Generator constraints:*
   The generator bus voltages as well as the real and reactive power generation must be limited by their lower and upper limits:

$$P_{Gi}^{min} < P_{Gi} < P_{Gi}^{max}, i = 1, \dots, N_G \quad (5)$$

$$Q_{Gi}^{min} < Q_{Gi} < Q_{Gi}^{max}, i = 1, \dots, N_G \quad (6)$$

$$V_{Gi}^{min} < V_{Gi} < V_{Gi}^{max}, i = 1, \dots, N_G \quad (7)$$

where $N_G$ is the number of generators.

2. *Transformer tap setting:*
   The transformer tap setting must be within their minimum and maximum limits:

$$T_i^{min} < T_i < T_i^{max}, i = 1, \dots, N_T \quad (8)$$

where $N_T$ is the number of transformers.

3. *Reactive compensators:*
   Shunt VARs are restricted by their boundaries as below:

$$Q_{Ci}^{min} < Q_{Ci} < Q_{Ci}^{max}, i = 1, \dots, N_C \quad (9)$$

where $N_C$ is the number of shunt compensators.

## 3. NATURE INSPIRED COMPUTING TECHNIQUES

### A. Gray Wolf Optimizer (GWO)

Gray wolf optimizer (GWO) is the latter nature inspired meta-heuristic technique first introduced by [15] which mimics the leadership and hunting mechanisms of gray wolves. Naturally, gray wolves are the top level predator in the food chain and live in a group of 5-12 wolves on average. Within the group, strict dominant hierarchy is practiced where the group is guides by the alphas, followed by betas, deltas and lastly omegas which are the lowest ranking of gray wolves that act as scapegoat. GWO is exploited based upon four stages: social hierarchy, tracking, encircling and attacking prey.

1. *Social hierarchy:*
   In order to mathematically model the GWO algorithm, alpha ($\alpha$) is considered as the fittest solution. The second and the third best solutions are assumed as beta ($\beta$) and delta ($\delta$) respectively. Consequently, the rest of the candidate solutions as omega ($\omega$). The hunting (optimization) in GWO algorithm is leaded by $\alpha$, $\beta$ and $\delta$ whereas the $\omega$ just following these three wolves.

2. *Encircling prey:*
   During hunting, the wolves encircling their prey which can be mathematically denoted as follows [15]:

$$\vec{D} = |\vec{C}.\vec{X_P}(t) - \vec{X}(t)| \quad (10)$$

$$\vec{X}(t+1) = \vec{X_P}(t) - \vec{A}.\vec{D} \quad (11)$$

where $\vec{A}$ and $\vec{C}$ are coefficient vectors, $\vec{X}$ and $\vec{X_P}$ are position vector of gray wolf and prey respectively, $t$ is the current iteration where the coefficient vectors are calculated as follows [15]:

$$\vec{A} = 2\,\vec{a}.\vec{r_1} - \vec{a} \quad (12)$$

$$\vec{C} = 2.\vec{r_2} \quad (13)$$

where $\vec{r_1}$ and $\vec{r_2}$ are random vectors in [0,1] while $\vec{a}$ are formulated to decreased from 2 to 0 over the course of iterations.

3. *Hunting:*
During hunting, the three best solutions are saved and the other search agents including omega wolves update their position according to the best position as expressed by the following formulas:

$$\overrightarrow{D_\alpha} = |\ \overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}\ |, \overrightarrow{D_\beta} = |\ \overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}\ |, \overrightarrow{D_\delta} = |\ \overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}\ | \tag{14}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.(\overrightarrow{D_\alpha}), \overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.(\overrightarrow{D_\beta}), \overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \tag{15}$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{16}$$

4. *Exploitation and Exploration:*
In (12) the parameter $\vec{a}$ can be formulated as exploitation and exploration processes. The candidate solutions (gray wolves) are converged toward the prey if $|\ \vec{A}\ | < 1$ and diverged from the prey if $|\ \vec{A}\ | > 1$.

*B. Ant Lion Optimizer (ALO)*

Ant lion optimizer (ALO) is another nature inspired computing technique which is also developed by Seyedali Mirjalili [16]. ALO algorithm mimics the hunting behavior of antlions in catching ants. ALO is created according to five main steps: random walks of ants, building pits, entrapment of ants, catching preys and rebuilding traps.

1. *Random walks of ants:*
Naturally, ants move randomly when searching and hunting for food which the ants' movement can be mathematically modeled as below [16]:

$$X(t) = [0, cumsum(2r(t_1) - 1), cumsum(2r(t_2) - 1), \ldots, cumsum(2r(t_n) - 1)] \tag{17}$$

where $t$ is the iteration, $cumsum$ is the cumulative sum, $n$ is the maximum number of iteration, and $r(t)$ is the stochastic function express as below:

$$r(t) = \begin{cases} 1\ if\ rand\ > 0.5 \\ 0\ if\ rand\ \ll 0.5 \end{cases} \tag{18}$$

where rand is a random number generated in between 0 and 1 uniformly. In order to keep the random walks of ants within the search boundary since ants will update its position during each optimization, (17) is normalized using the min-max normalization as follows:

$$X_i^t = \frac{(X_i^t - a_i) \times (d_i - c_i^t)}{(d_i^t - a_i)} + c_i \tag{19}$$

where $a_i$ is the minimum of random walk of i-th variable, $c_i$ and $d_i$ are the minimum and maximum of all variables for i-th ant respectively, $c_i^t$ and $d_i^t$ are the minimum and maximum of i-th variable at t-th iteration respectively.

2. *Trapping in antlions' pits:*
The effect of antlions' pits on random walks of ants is mathematically modeled as below [16]:

$$c_i^t = Antlion_j^t + c^t \tag{20}$$

$$d_i^t = Antlion_j^t + d^t \tag{21}$$

where $Antlion_j^t$ is the position of the selected j-th antlion at t-th iteration, $c^t$ and $d^t$ are the minimum and maximum of all variables at t-th iteration.

3. *Building pits:*
ALO algorithm utilized roulette wheel operator for choosing antlions based on their fitness during optimization. This operator gives higher opportunity to the antlions for trapping ants.

4. *Sliding ants against towards antlions:*
   Once the antlion realizes a prey is in the trap, they will throw the sand outward the center of the pit to slide the ant into the bottom of the trap. This behavior can be modeled as below [16]:

$$c^t = \frac{c^t}{I} \tag{22}$$

$$d^t = \frac{d^t}{I} \tag{23}$$

   where I is the ratio.

5. *Catching preys and rebuilding the traps:*
   The last step of this mechanism is when an ant is being caught by antlion in the trap where the ant becomes fitter than its predator. Consequently, the antlion will update its current position of the hunted ant to enhance its chance of catching new prey which can be expressed as follows [16]:

$$Antlion_j^t = Ant_i^t \ if \ f(Ant_j^t) > f(Antlion_j^t) \tag{24}$$

   where $Ant_i^t$ and $Ant_j^t$ are the position of the selected i-th and j-th ant at t-th iteration respectively.

6. *Elitism:*
   Elitism is vital for them to maintain the best solutions obtained for each optimization process as the movement of the ants can be affected by the fittest antlion (elite). Hence, each ant randomly walks around a selected antlion by the roulette wheel and as well the elite simultaneously as expressed as below [16]:

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \tag{25}$$

   where $R_A^t$ is the random walk around the antlion selected by the roulette wheel at t-th iteration and $R_E^t$ is the random walk around the elite at t-th iteration.

## 5. SIMULATION AND DISCUSSION

The two nature inspired computing techniques are implemented on IEEE-30 bus system with 13 control variables [8] and the results are compared between each other. This system comprising of 6 generators, 41 transmission lines, 4 transformers and 3 reactive compensation devices placed at buses 3, 10 and 24 respectively. The lower and upper limits for the control variables such as generators voltages, transformer tap settings and reactive compensation devices are depicted in Table 1. In this case study, the load demand is set as below:

$$P_{Load} = 2.832 \ p.u \qquad Q_{Load} = 1.262 \ p.u$$

The best solution of GWO and ALO in solving ORPD is depicted in Table 2. It is obvious that the optimal results obtained by GWO yields a lower power loss than that obtained by using ALO. Comparison of the loss obtained between GWO and ALO is only about 0.38% of loss reduction.

Fig.1 exhibited the performance of GWO in term of best score vs. iteration. From this figure, it can be noted that the best result is 4.625 and the optimization processed in 155.2054 sec whereas Fig. 2 illustrated the performance of ALO over 150 iterations and the best result obtained by ALO is 4.6694. The optimization for ALO processed in 156.8746 sec. Furthermore, Fig. 3 demonstrated the comparison between the performance of GWO and ALO.

## 6. CONCLUSION

The nature inspired computing techniques namely gray wolf optimizer and ant lion optimizer in overcoming ORPD problem is presented in this paper. The performance of GWO and ALO are evaluated using IEEE 30-bus system and the results obtained are compared between these two methods. The simulation results shows that GWO is able to obtained better results as compared with ALO.

Table 1: Limit setting for the control variables of IEEE-30 bus system.

| **Control Variables** | **Limits** | |
|---|---|---|
| | **Lower Limit** | **Upper Limit** |
| Generator Voltages | 0.9 p.u | 1.1 p.u |
| Transformer Tap Settings | 0.95 p.u | 1.05 p.u |
| Reactive Compensation Devices | -12 MVar | 36 MVar |

Table 2: Comparison of ORPD results of control variables between GWO and ALO.

| **Control Variables** | **Nature Inspired Computing Techniques** | |
|---|---|---|
| | **GWO [14]** | **ALO [17]** |
| $V_1$ | 1.1000 | 1.1000 |
| $V_2$ | 1.096149 | 1.0948 |
| $V_5$ | 1.080036 | 1.0759 |
| $V_8$ | 1.080444 | 1.0774 |
| $V_{11}$ | 1.093452 | 1.0761 |
| $V_{13}$ | 1.1000 | 1.1000 |
| $T_1$ | 1.04 | 1.03 |
| $T_2$ | 0.95 | 1.00 |
| $T_3$ | 0.95 | 1.01 |
| $T_4$ | 0.95 | 0.98 |
| $Q_1$ | 12 | -1 |
| $Q_2$ | 30 | 25 |
| $Q_3$ | 8 | 11 |
| Loss (MW) | 4.5984 | 4.616 |

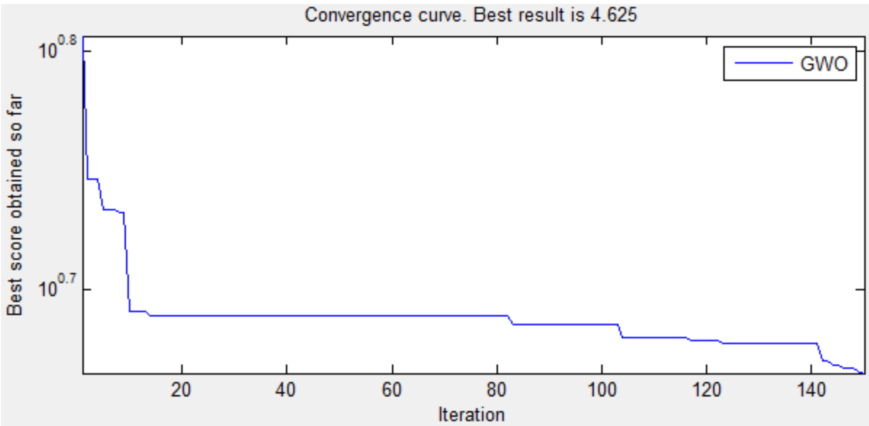


Figure 1: Convergence curve of the performance of GWO.

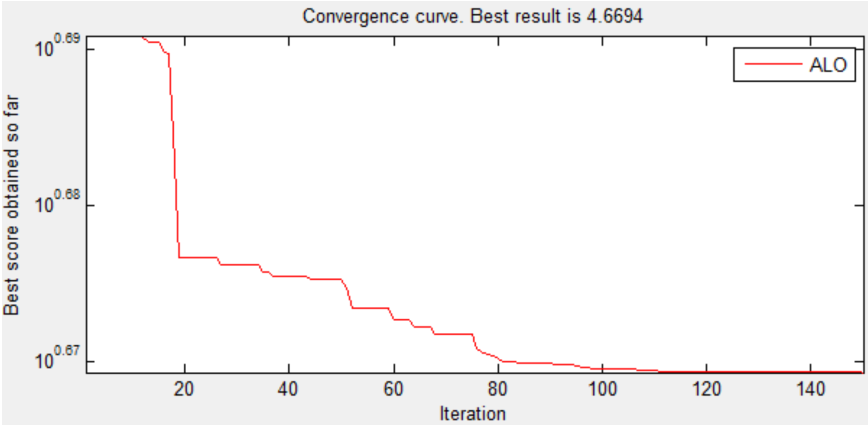


Figure 2: Convergence curve of the performance of ALO.

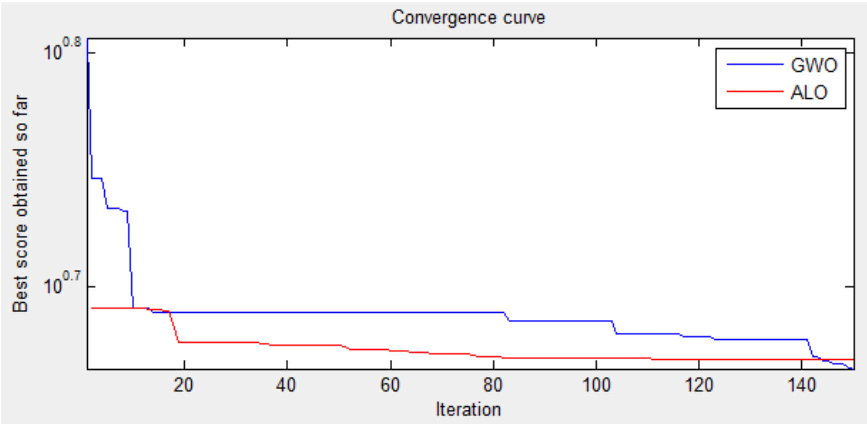


Figure 3: Comparison of convergence curve of the performance of GWO and ALO.

## References


[1] M. Varadarajan and K. S. Swarup, "Differential evolution approach for optimal reactive power dispatch," *Applied Soft Computing,* vol. 8, pp. 1549-1561, 9// 2008.

[2] E. Acha, H. Ambriz-Perez, and C. R. Fuerte-Esquivel, "Advanced transformer control modeling in an optimal power flow using Newton's method," *Power Systems, IEEE Transactions on,* vol. 15, pp. 290-298, 2000.

[3] D. S. Kirschen and H. P. Van Meeteren, "MW/voltage control in a linear programming based optimal power flow," *Power Systems, IEEE Transactions on,* vol. 3, pp. 481-489, 1988.

[4] I. Perez Abril and J. A. G. Quintero, "VAr compensation by sequential quadratic programming," *Power Systems, IEEE Transactions on,* vol. 18, pp. 36-41, 2003.

[5] E. M. Soler, E. N. Asada, and G. R. M. da Costa, "Penalty-Based Nonlinear Solver for Optimal Reactive Power Dispatch With Discrete Controls," *Power Systems, IEEE Transactions on,* vol. 28, pp. 2174-2182, 2013.

[6] P. Subbaraj and P. N. Rajnarayanan, "Optimal reactive power dispatch using self-adaptive real coded genetic algorithm," *Electric Power Systems Research,* vol. 79, pp. 374-381, 2// 2009.

[7] K. Ayan and U. Kılıç, "Artificial bee colony algorithm solution for optimal reactive power flow," *Applied Soft Computing,* vol. 12, pp. 1477-1482, 5// 2012.

[8] A. H. Khazali and M. Kalantar, "Optimal reactive power dispatch based on harmony search algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 33, pp. 684-692, 3// 2011.

[9] N. Sinsuphan, U. Leeton, and T. Kulworawanichpong, "Optimal power flow solution using improved harmony search method," *Applied Soft Computing,* vol. 13, pp. 2364-2374, 5// 2013.

[10] B. Zhao, C. X. Guo, and Y. J. Cao, "A multiagent-based particle swarm optimization approach for optimal reactive power dispatch," *Power Systems, IEEE Transactions on,* vol. 20, pp. 1070-1078, 2005.

[11] J. G. Vlachogiannis and K. Y. Lee, "A Comparative Study on Particle Swarm Optimization for Optimal Steady-State Performance of Power Systems," *Power Systems, IEEE Transactions on,* vol. 21, pp. 1718-1728, 2006.

[12] S. Duman, U. Güvenç, Y. Sönmez, and N. Yörükeren, "Optimal power flow using gravitational search algorithm," *Energy Conversion and Management,* vol. 59, pp. 86-95, 7// 2012.

[13] B. Shaw, V. Mukherjee, and S. P. Ghoshal, "Solution of reactive power dispatch of power systems by an opposition-based gravitational search algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 55, pp. 29-40, 2// 2014.

[14] M. H. Sulaiman, Z. Mustaffa, M. R. Mohamed, and O. Aliman, "Using the gray wolf optimizer for solving optimal reactive power dispatch problem," *Applied Soft Computing,* vol. 32, pp. 286-292, 7// 2015.

[15] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software,* vol. 69, pp. 46-61, 3// 2014.

[16] S. Mirjalili, "The Ant Lion Optimizer," *Advances in Engineering Software,* vol. 83, pp. 80-98, 5// 2015.

[17] R. N. S. Mei, M. H. Sulaiman, Z. Mustaffa, "Ant Lion Optimizer for Optimal Reactive Power Dispatch Solution", Journal of Electrical Systems "Special Issue AMPE2015", pp. 68-74.